

Desenvolupament de llocs web amb PHP i MySQL

TEMA 2. TEORIA 1

FORMULARIS. MANEIG DE FITXERS. PAS DE VARIABLES. SERVIDOR

1. FORMULARIS

1.1. ELEMENTS D'ENTRADA A FORMULARI. ACCÉS DES DE PHP

1.2. PAS DE DADES A UN FORMULARI

1.2.1. Amb variables globals

1.2.2. Amb arrays associatius

1.2.3. Per la URL

1.2.4. Pas de Matrius A PHP

1.2.5. Per QUERY STRING

1.3. PROCESSAMENT DE FORMULARIS EN PHP

1.3.1. En un sol script. FORMULARIS REENTRANTS

1.3.2. Validació de formularis des dels ervidor

2. MANEIG DE FITXERS

3. MANEIG DE DIRECTORIS

4. EXECUCIÓ DE COMANDAMENTS DEL SISTEMA OPERATIU EN PHP

5. ENVIAMENT DE FITXERS AL SERVIDOR

6. ENLLAÇOS D'INTERÉS

1. FORMULARIS

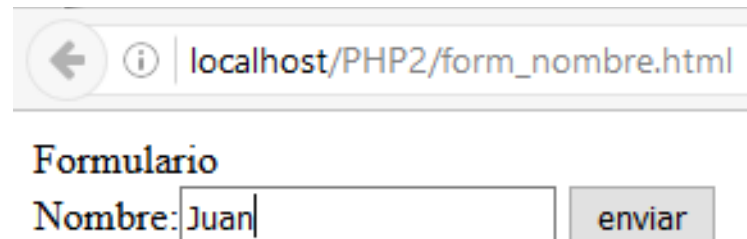
El llenguatge PHP ens proporciona una manera senzilla de manejar formularis, permetent-nos processar la informació que l'usuari ha introduït.

Un formulari fa de cada objecte (caixes de text, botons de radi, etc.) una variable, que tindrà aparellat una dada.

Suposem el següent formulari (arxiu **form_nom.html**):

```
<html>
  <body>
    <div>Formulario Nombre</div>
    <form action="form_nombre.php" method="get">
      nombre:<input type="text" name="nombre" value="" size="20">
      <input type="submit" name="enviar" value="enviar">
    </form>
  </body>
</html>
```

En executar-lo tindrem:



Formulario

Nombre:

Quan l'usuari prem el botó d'enviament d'un formulari (*SUBMIT*) es mana al servidor una cadena de la forma:

nom=valor&nom=valor&nom=valor...

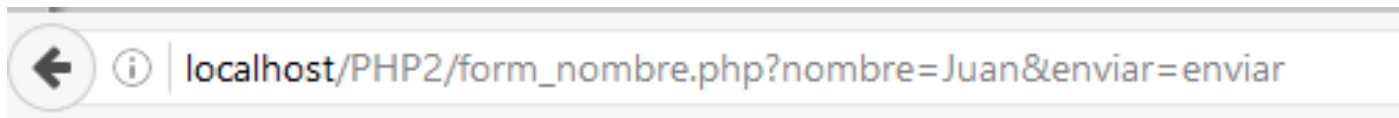
La manera de rebre'ls dependrà d'atribut **METHOD** que hàgem posat en el FORM:

Exemple:

Si enviem les dades mitjançant **GET**, es processaran pel següent script **processa_formulari.php**: i podem observar que els parells *nom-valor* van en la **URL**:

http://localhost/PHP2/form_nom.php?nom=juan&enviar=enviar

En prémer *Enviar* veurem l'eixida:



hola Juan

Mètode GET:

- Les dades s'envien en un missatge **HTTP-GET** que conté la cadena amb els parells **nom=valor** precedits del caràcter '?' a la **direcció URL** amb el la sol·licitud del script que tractarà les dades.
- Aquesta cadena és denominada **QUERY_STRING** i si usem un CGI s'emmagatzemarà en una variable de entorno amb aqueix nom en el servidor.
- PHP emmagatzemarà aquests valors en la matriu **superglobal \$_GET** utilitzant com a índex de cerca del valor el nom **\$_GET['nom']**.
- També podem usar la matriu **\$_REQUEST** per a accedir a un valor rebut mitjançant HTTP-GET.

Mètode POST:

- Les dades s'envien en un missatge **HTTP-POST** on es passen els parells **nom-valor** en **el propi cos del missatge de sol·licitud** http i **no en la URL**.
- PHP emmagatzemarà aquests valors en la matriu superglobal **\$_POST** utilitzant com a índex de cerca del valor el nom **\$_POST['nom']**.
- Haurem d'usar aquest estil intermedi de referència i no l'estil curt i llarg, tal com comentàvem en el GET.
- També podem usar la matriu **\$_REQUEST** per a accedir a un valor rebut mitjançant HTTP-POST.

Si recordem, l'etiqueta `html` del formulari (`<form>`) té diversos atributs que ens interessen:

- `name`: nom del formulari
- `method`: defineix la modalitat d'enviament de la informació. Els valors possibles són **POST** o **GET**;
- `action`: indica l'acció o el destí per a les dades del formulari, en el nostre cas apuntarà al fitxer `php` que manipularà aquestes dades
- `enctype`: especifica el tipus d'enciptació que es realitzarà amb les dades que s'enviaran. Aquest atribut solament s'aplica si `method` és **POST**.

Nota:

Per defecte és "`application/x-www-form-urlencoded`".

Un altre possibles valor és "`multipart/form-data`" que utilitzem quan tenim en el formulari camps de tipus fitxer.

- `target`: nom del marc on es visualitzarà la informació retornada pel servidor. Funciona igual que en l'element ancora d'hipertext A.

L'acció de prémer el botó d'enviament del formulari desencadena la transmissió de certs a dades a una pàgina receptora (**`action=pàgina.php`**), a una adreça de correu electrònic (**`action=mailto:micorreo@correo.es`**) o un programa o CGI per al seu tractament.

PHP entén totes les dades d'un formulari com un ***array associatiu***, de manera que podrem utilitzar amb ell funcions de arrays.

exemple: `count($_POST)`.

Funcions php relacionades amb formularis:

Un dels problemes que podem tindre quan usem formularis és el tipus de dades amb què els usuaris emplenaran els camps.

Poden introduir caràcters estranys. Referent a això hi ha diverses funcions *de neteja* que ens poden ajudar:

- `htmlspecialchars($variable)` → converteix els signes `<`, `>`, `&` i `"` en les seues corresponents entitats html (`<`, `>`, `&Amp;`, `"`)
- `striplashes($variable)` → elimina totes les barres invertides d'una cadena
- `nl2br($variable)` → converteix tots els salts de línia de l'editor en l'equivalent HTML
- `isset($variable)` → Si no emplenem tots els camps d'un formulari i premem el botó d'Enviar, podem tindre errors. En aquest cas és útil la funció `isset($variable)`, que comprova si la variable s'ha definit (*true*) o no (*false*).
- `empty($variable)` → serveix per a comprovar si una variable no està assignada, si està buida o si correspon a valor 0.

Exemple complet:

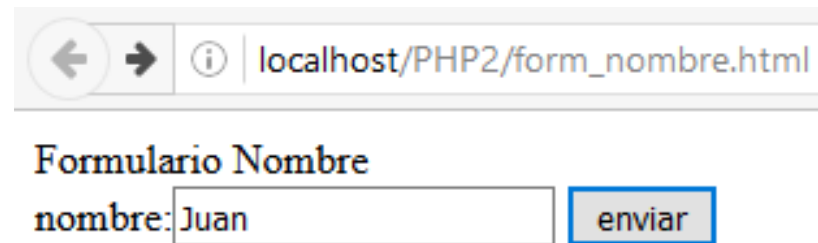
Form_Nom.html:

```
<body>
  <div>Formulario Nombre</div>
  <form action="form_nombre.php" method="get">
    nombre:<input type="text" name="nombre" value="" size="20">
    <input type="submit" name="enviar" value="enviar">
  </form>
</body>
```

Form_nom.php:

```
if (!empty ($_GET['nombre']))
    echo 'hola '.$_GET['nombre'];
else
    echo 'no hay datos para visualizar';
```

Execució:

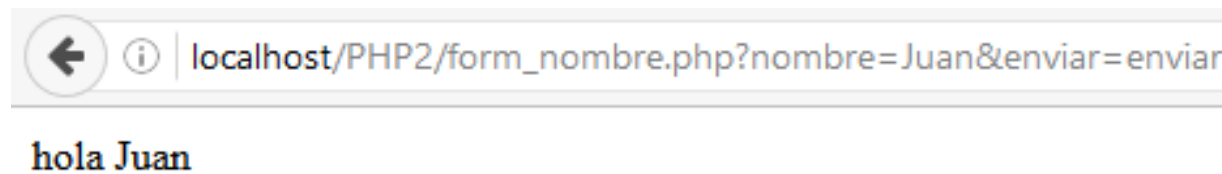


← → ⓘ localhost/PHP2/form_nombre.html

Formulario Nombre

nombre:

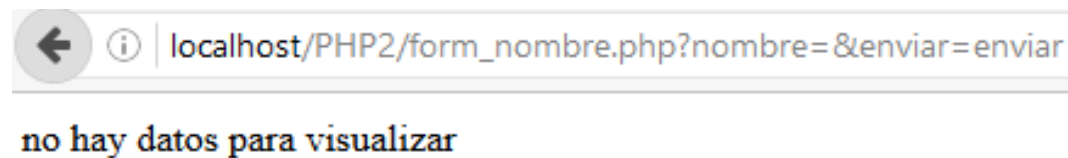
Introduïm Nom i premem Enviar:



← ⓘ localhost/PHP2/form_nombre.php?nombre=Juan&enviar=enviar

hola Juan

Si no Introduïm Nom i premem Enviar, veurem:



← ⓘ localhost/PHP2/form_nombre.php?nombre=&enviar=enviar

no hay datos para visualizar

Nota:

Per a comprovar si una variable d'un formulari té assignat valor també es pot fer:

```
<?php
if (!empty($_POST["comentario"])){
    $comment=htmlspecialchars($_POST["comentario"]);
    $comment=stripslashes($comment);
    $comment=nl2br($comment);
    echo "Has escrito:".$comment;
}
?>
```

1.1 ELEMENTS D'ENTRADA A FORMULARI. ACCÉS DES DE PHP

Un formulari permet els següents elements per a la entrada d'informació :

- Elementos de tipo INPUT
 - TEXT
 - RADIO
 - CHECKBOX
 - BUTTON
 - FILE
 - HIDDEN
 - PASSWORD
 - SUBMIT
- Elemento SELECT
 - Simple / múltiple
- Elemento TEXTAREA

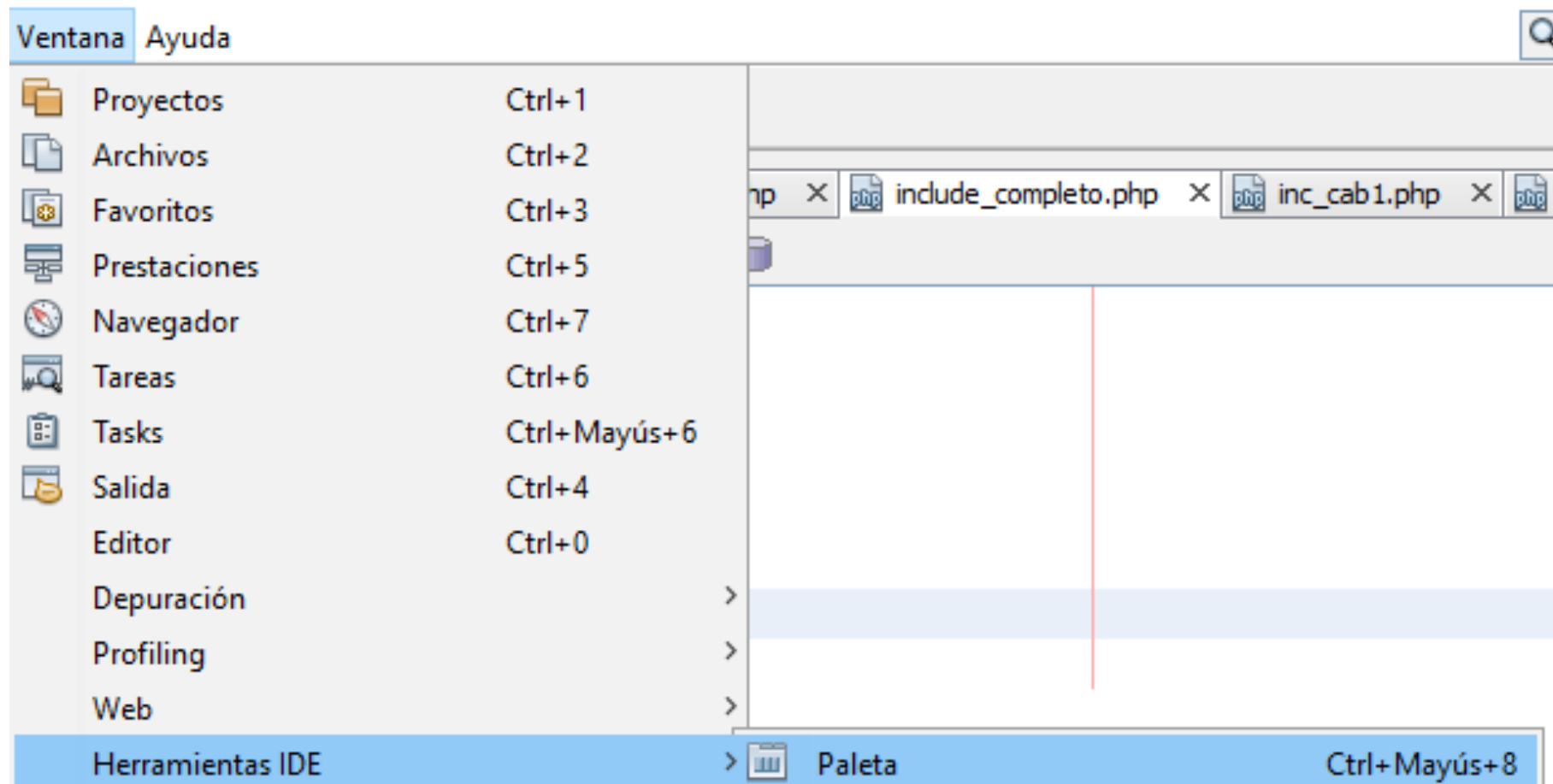
Nota:

Existeixen moltes eines per a facilitar-nos el disseny i generació de codi html, però nosaltres usarem:

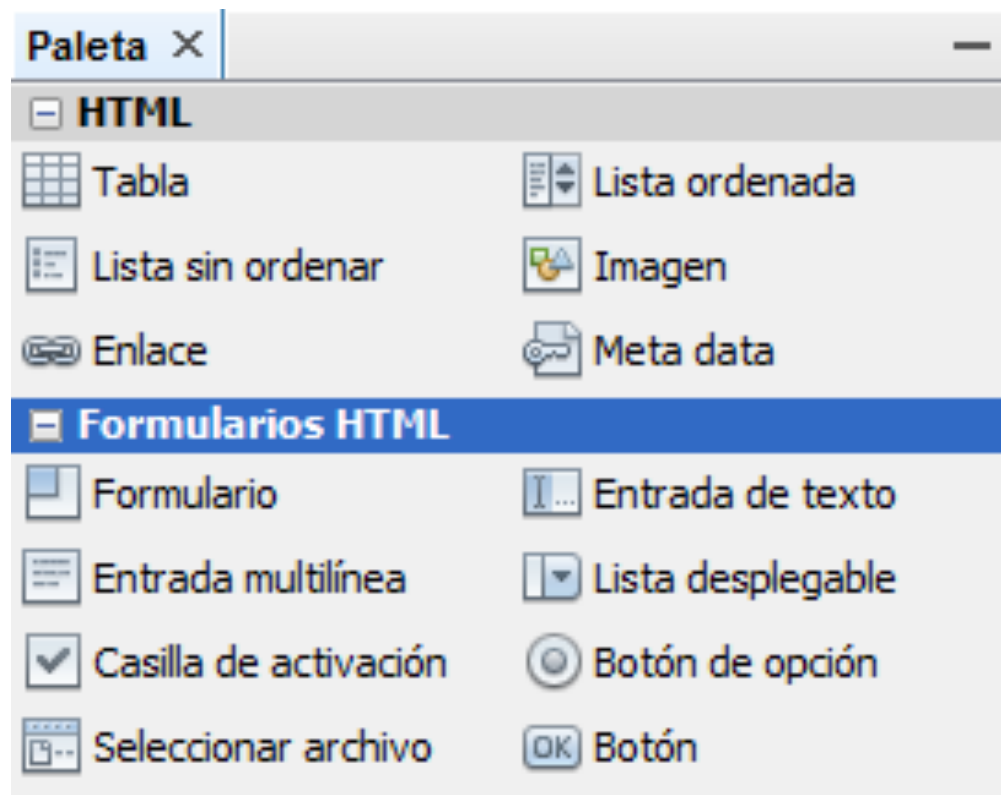
- NETBEANS
- DreamWeaver
- CKeditor4

COM INCLOURE ELEMENTS DE FORMULARI DES DE DES DE NETBEANS:

Des de FINESTRA - EINES IDE – PALETA:



Ens mostra una finestra com la següent:



Ara, mitjançant **Seleccionar- Arrossegar i soltar**, podem incloure cadascun d'aqueixos elements en la nostra aplicació

Nota:

Per a tots els exemples d'aquest apartat (Elements d'Entrada a formulari) que venen a continuació cal substituir `$_request[]` per `$_POST[]` o `$_GET[]`.

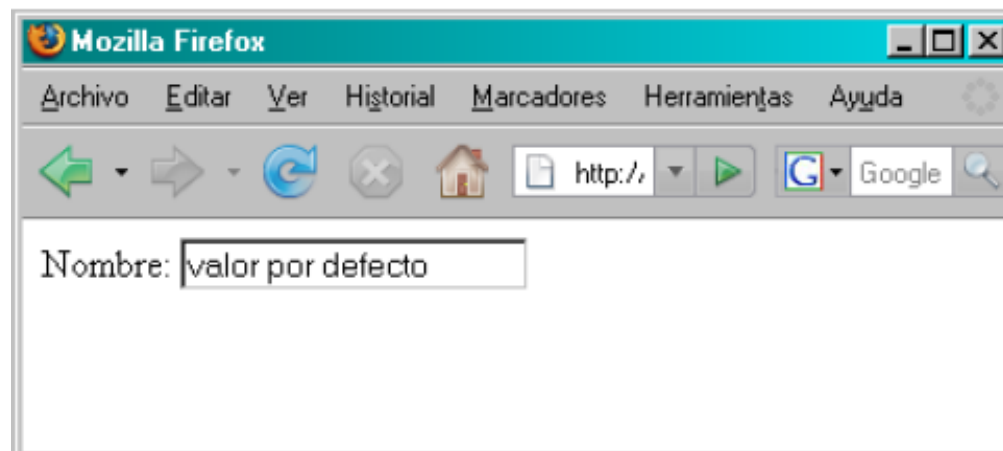
Això és pel fet que `$_request[]` és una variable GLOBAL i és accessible des de qualsevol part del script.

A partir de la PHP 5.4.0 només es pot accedir a dades del formularis HTML usant `$_POST[]` o `$_GET[]`.

INPUT TEXT

```
html:
nombre:
<input type="text" name="cadena" value="valor por defecto" size="20">

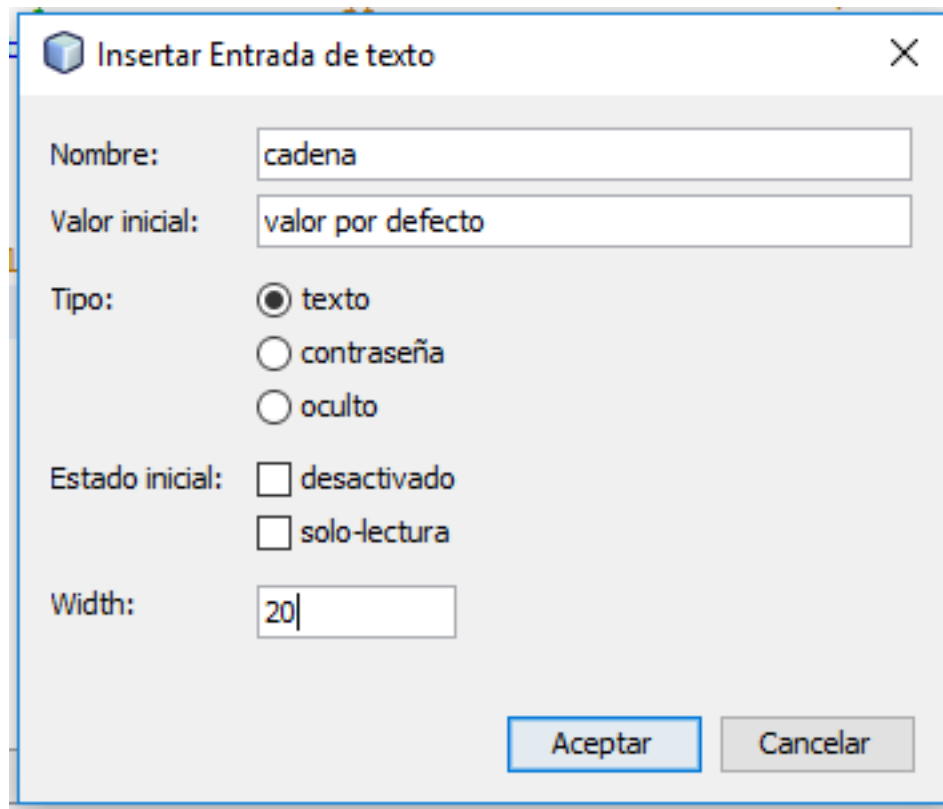
php:
<?php
$cadena = $_request['cadena'];
echo $cadena;
?>
```



Des de NETBEANS:

Veure Arxiu ***index2.php*** del PROJECTE PHP2:

PALETA - ENTRADA DE TEXT:



The image shows a dialog box titled "Insertar Entrada de texto" (Insert Text Input) with a close button (X) in the top right corner. The dialog contains the following fields and options:

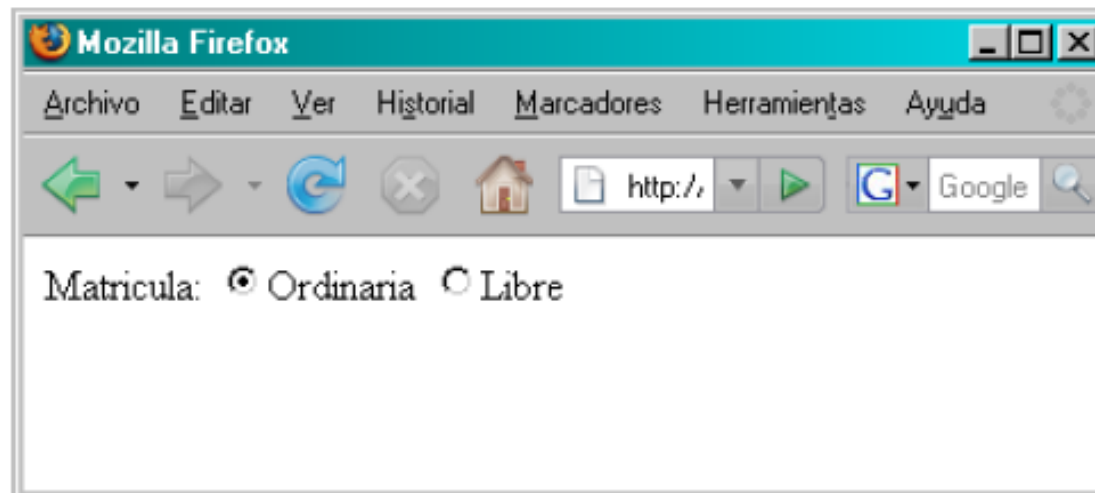
- Nombre:** A text field containing "cadena".
- Valor inicial:** A text field containing "valor por defecto".
- Tipo:** Three radio button options: "texto" (selected), "contraseña", and "oculto".
- Estado inicial:** Two checkbox options: "desactivado" and "solo-lectura", both of which are unchecked.
- Width:** A text field containing "20".

At the bottom right of the dialog are two buttons: "Aceptar" (Accept) and "Cancelar" (Cancel).

INPUT RÀDIO

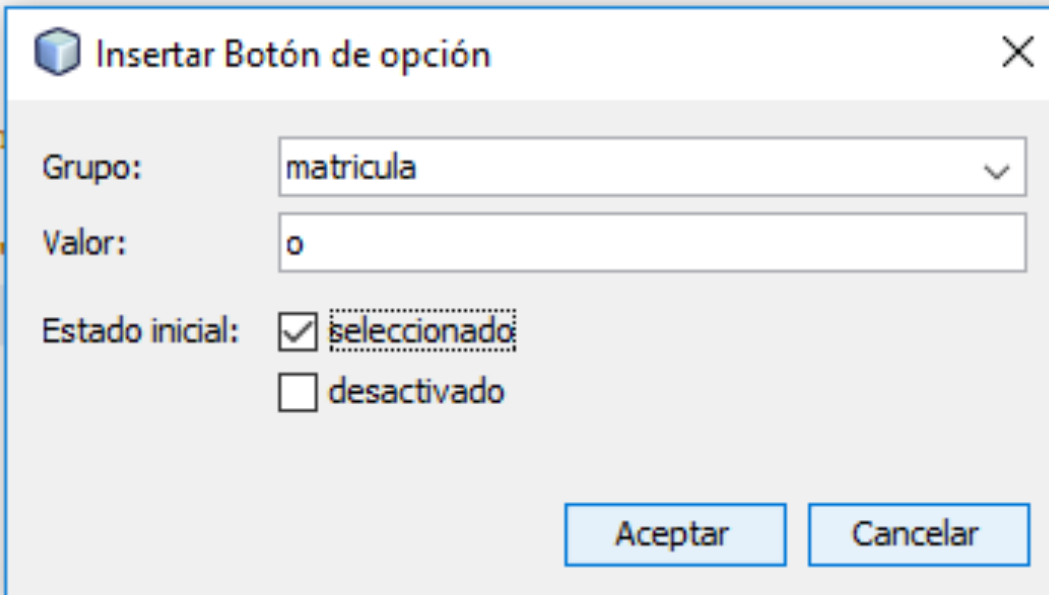
```
html:
matricula:
<input type="radio" name="matricula" value="0" checked>ordinaria
<input type="radio" name="matricula" value="1">libre

php:
<?php
$matricula = $_request['matricula'];
echo $matricula; // visualitzarà 0 o 1
?>
```



Des de NETBEANS:

PALETA - BOTÓ D'OPCIÓ:



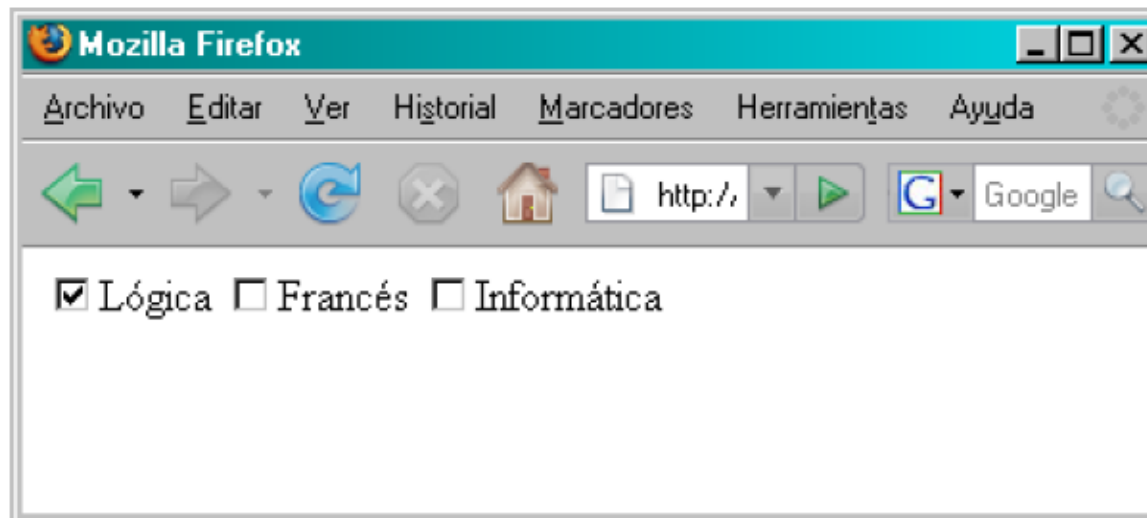
The image shows a dialog box titled "Insertar Botón de opción" with a close button (X) in the top right corner. The dialog contains the following fields and options:

- Grupo:** A dropdown menu with the value "matricula" selected.
- Valor:** A text input field containing the value "0".
- Estado inicial:** Two radio button options:
 - ☒ seleccionado
 - ☐ desactivado
- Buttons:** "Aceptar" and "Cancelar" buttons at the bottom right.

INPUT CASELLA DE SELECCIÓ

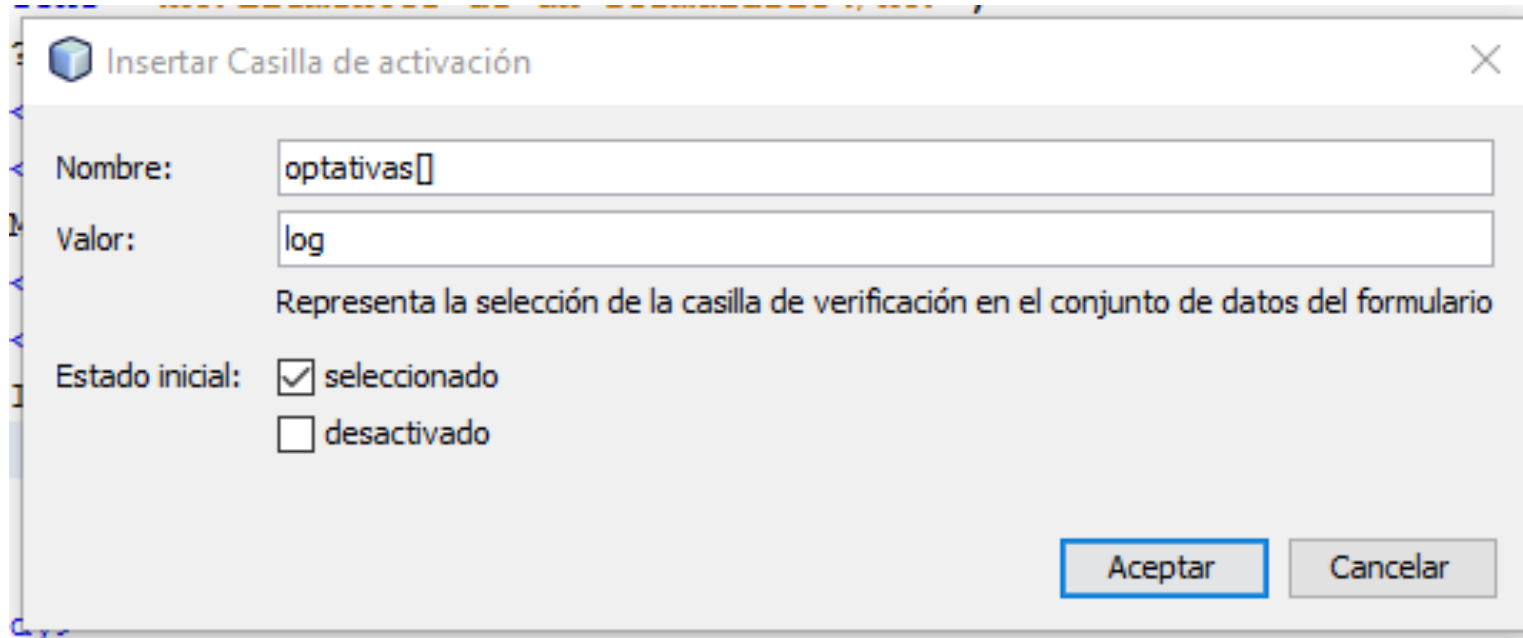
```
html:
<input type="checkbox" name="optativas[]" value="log" checked>lógica
<input type="checkbox" name="optativas[]" value="fra">francés
<input type="checkbox" name="optativas[]" value="inf">informática

php:
<?php
$optativas = $_request['optativas']; // devolverá una matriz.
foreach ($optativas as $optativa)
    echo $optativa.'<br>';
?>
```



Des de NETBEANS:

PALETA – CASELLA D'ACTIVACIÓ:



The image shows a dialog box titled "Insertar Casilla de activación" (Insert Activation Checkbox) with a close button (X) in the top right corner. The dialog contains the following fields and options:

- Nombre:** A text field containing "optativas".
- Valor:** A text field containing "log".
- Representa la selección de la casilla de verificación en el conjunto de datos del formulario** (Represents the selection of the verification checkbox in the form's data set).
- Estado inicial:** Two radio button options:
 - ☒ seleccionado (selected)
 - ☐ desactivado (disabled)
- Buttons:** "Aceptar" (Accept) and "Cancelar" (Cancel) at the bottom right.

INPUT BUTTON

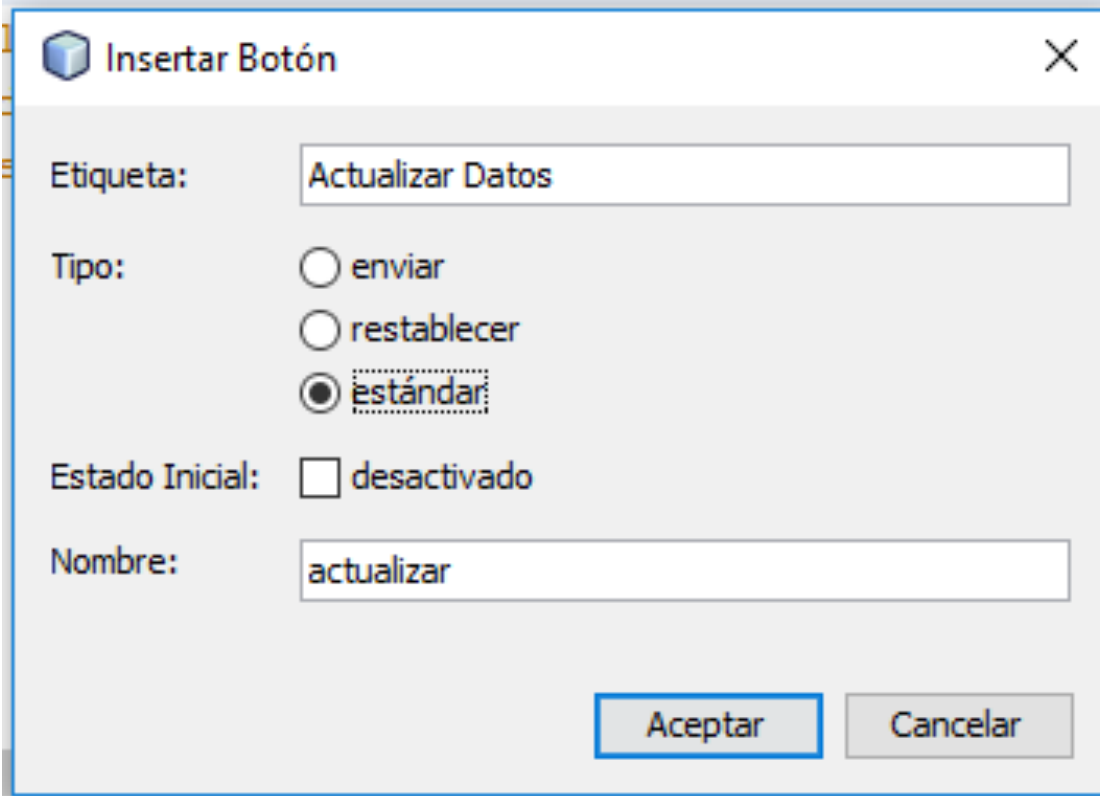
```
html:
<input type="button" name="actualizar" value="actualizar datos">

php:
<?php
$actualizar = $_request['actualizar'];
if (isset($actualizar) === true)
echo 'se han actualizado los datos';
?>
```



Des de NETBEANS:

PALETA – BOTÓ. Tipus estàndard:



Insertar Botón

Etiqueta:

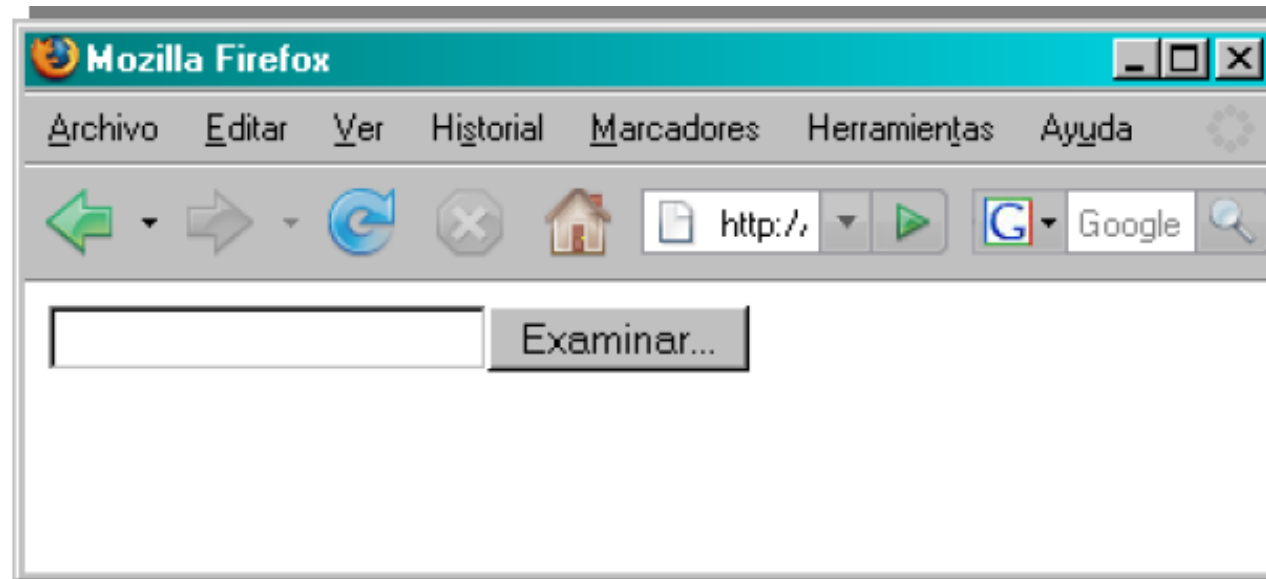
Tipo: ☐ enviar ☐ restablecer ☒ estándar

Estado Inicial: ☐ desactivado

Nombre:

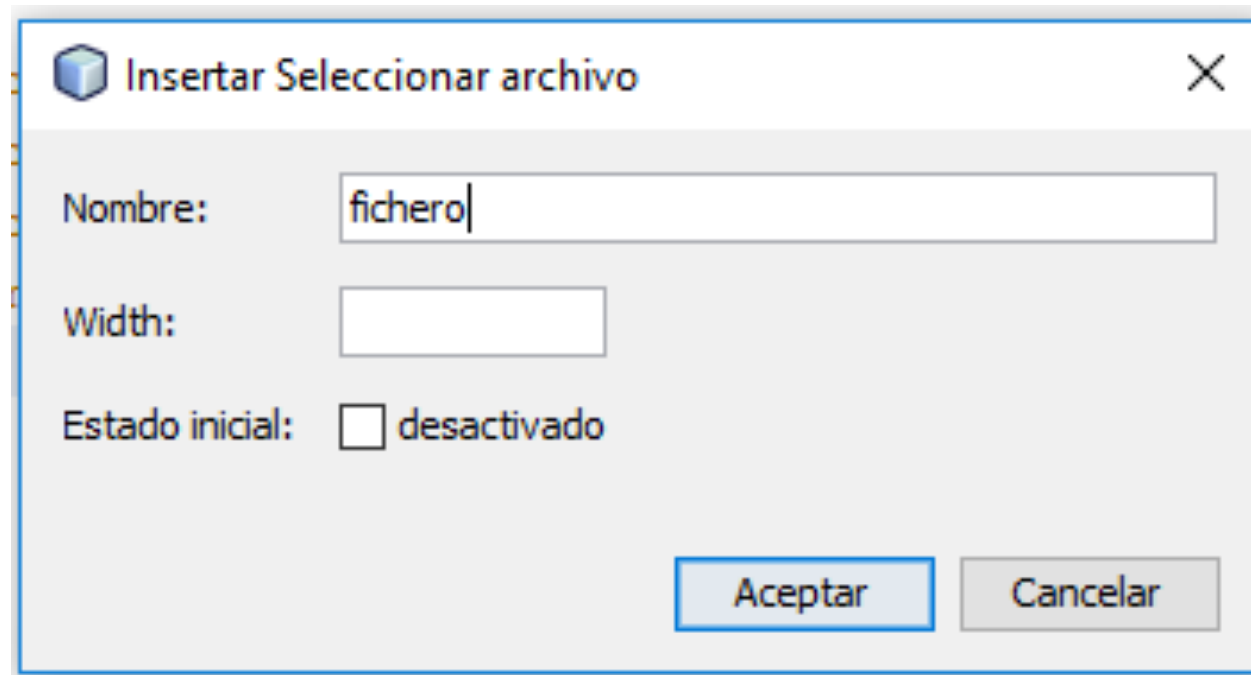
INPUT FILE

```
html:  
<form action="<?php echo $_SERVER['PHP_SELF'] ?>" method="post"  
  enctype="multipart/form-data">  
  <input type="file" name="fichero">  
</form>
```



Des de NETBEANS:

PALETA - SELECCIONAR ARXIU:



INPUT HIDDEN

html:

```
<input type="hidden" name="username" value="<?php $usuario ?>">
```

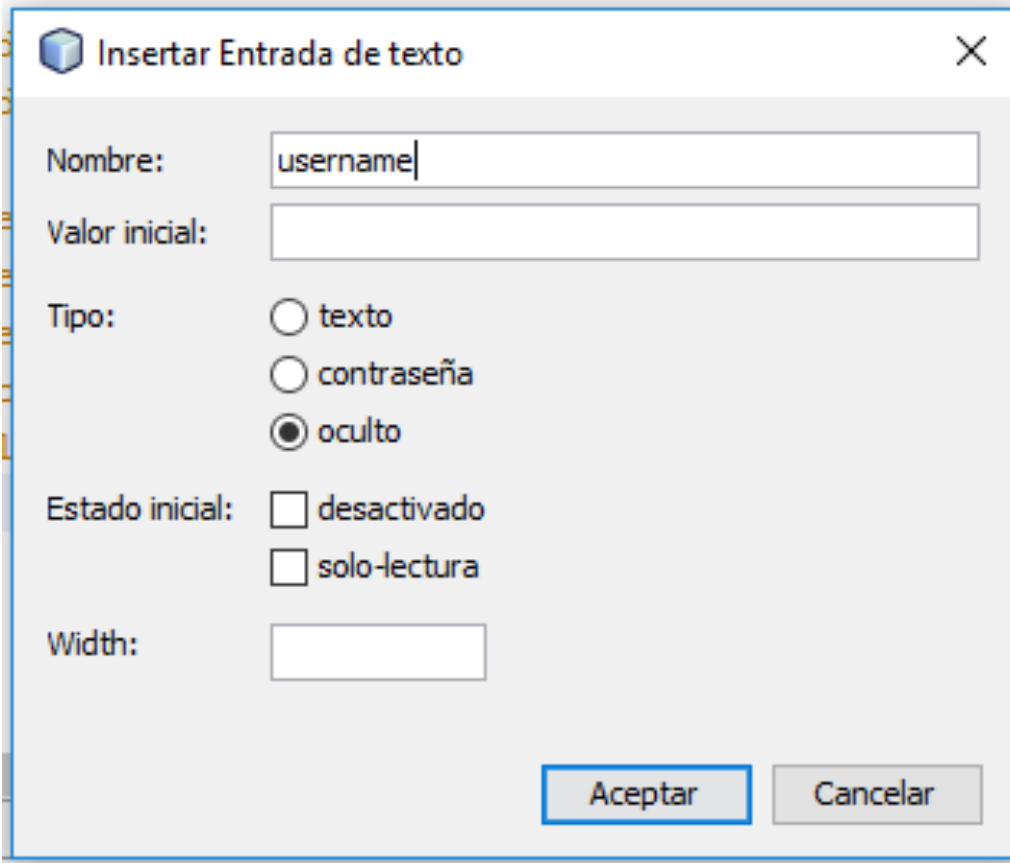
php:

```
<?php  
$username = $_request['username'];  
echo $username;  
?>
```

No es visualitzarà res.

Des de NETBEANS:

PALETA - ENTRADA DE TEXT. Tipus ocult:



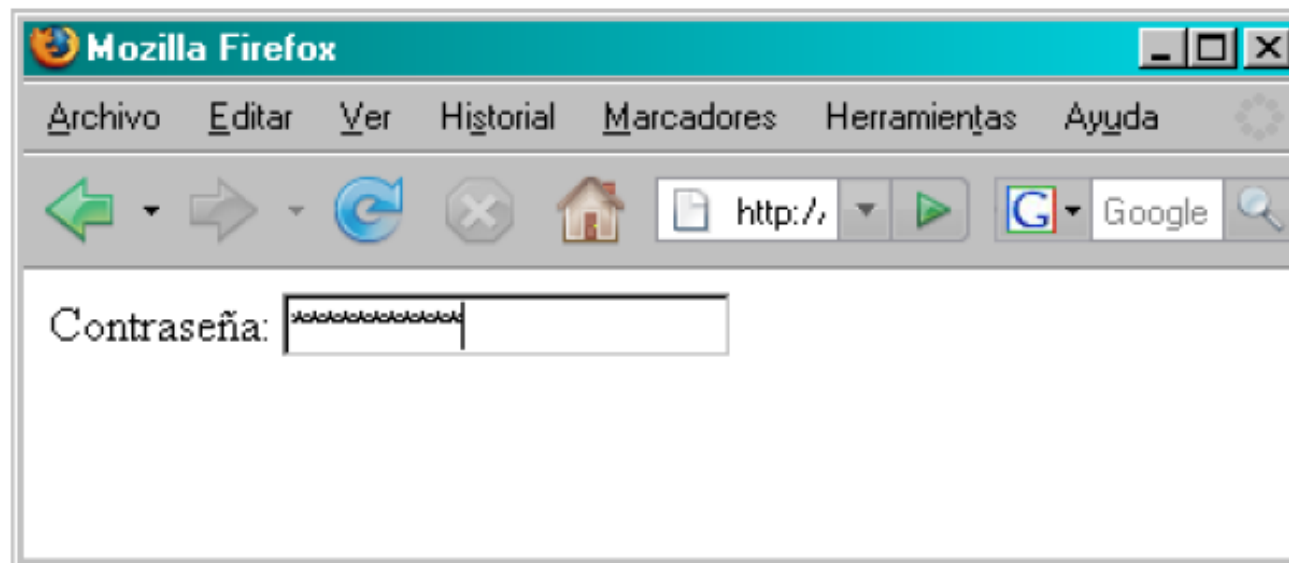
The image shows a dialog box titled "Insertar Entrada de texto" (Insert Text Input) with a close button (X) in the top right corner. The dialog contains the following fields and options:

- Nombre:** A text field containing the value "username".
- Valor inicial:** An empty text field.
- Tipo:** Three radio button options: "texto", "contraseña", and "oculto". The "oculto" option is selected.
- Estado inicial:** Two checkbox options: "desactivado" and "solo-lectura". Both are currently unchecked.
- Width:** An empty text field.

At the bottom right of the dialog are two buttons: "Aceptar" (Accept) and "Cancelar" (Cancel). The "Aceptar" button is highlighted with a blue border.

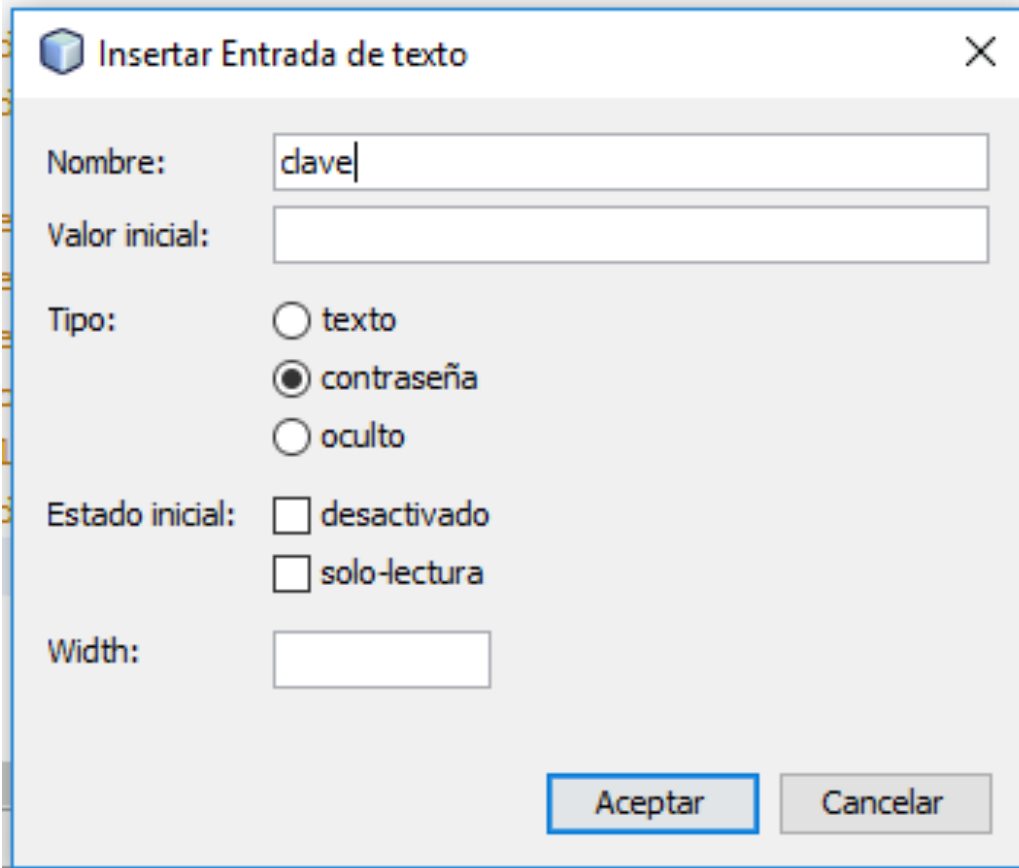
INPUT PASSWORD

```
html:  
contraseña: <input type="password" name="clave">  
  
php:  
<?php  
$clave = $_request['clave'];  
echo $clave;  
?>
```



Des de NETBEANS:

PALETA - ENTRADA DE TEXT. Tipus Contrasenya:



The image shows a NetBeans dialog box titled "Insertar Entrada de texto". It contains the following fields and options:

- Nombre:** A text field containing the word "clave".
- Valor inicial:** An empty text field.
- Tipo:** Three radio button options: "texto", "contraseña" (which is selected), and "oculto".
- Estado inicial:** Two checkbox options: "desactivado" and "solo-lectura", both of which are unchecked.
- Width:** An empty text field.
- Buttons:** "Aceptar" and "Cancelar" buttons at the bottom right.

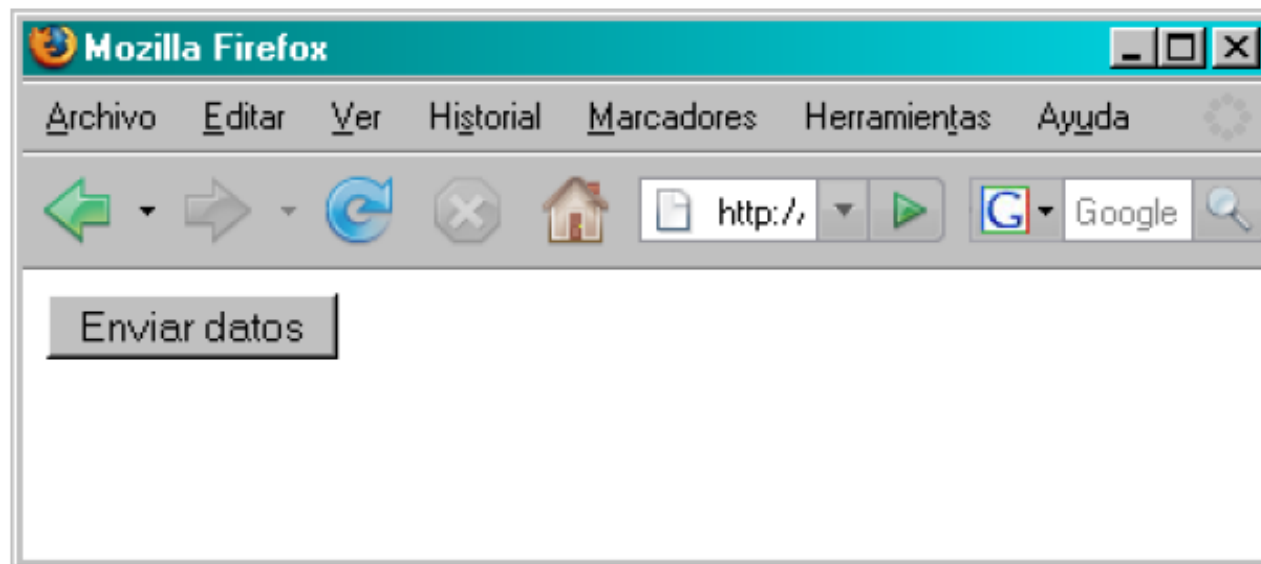
INPUT SUBMIT

[html:](#)

```
<input type="submit" name="enviar" value="enviar datos">
```

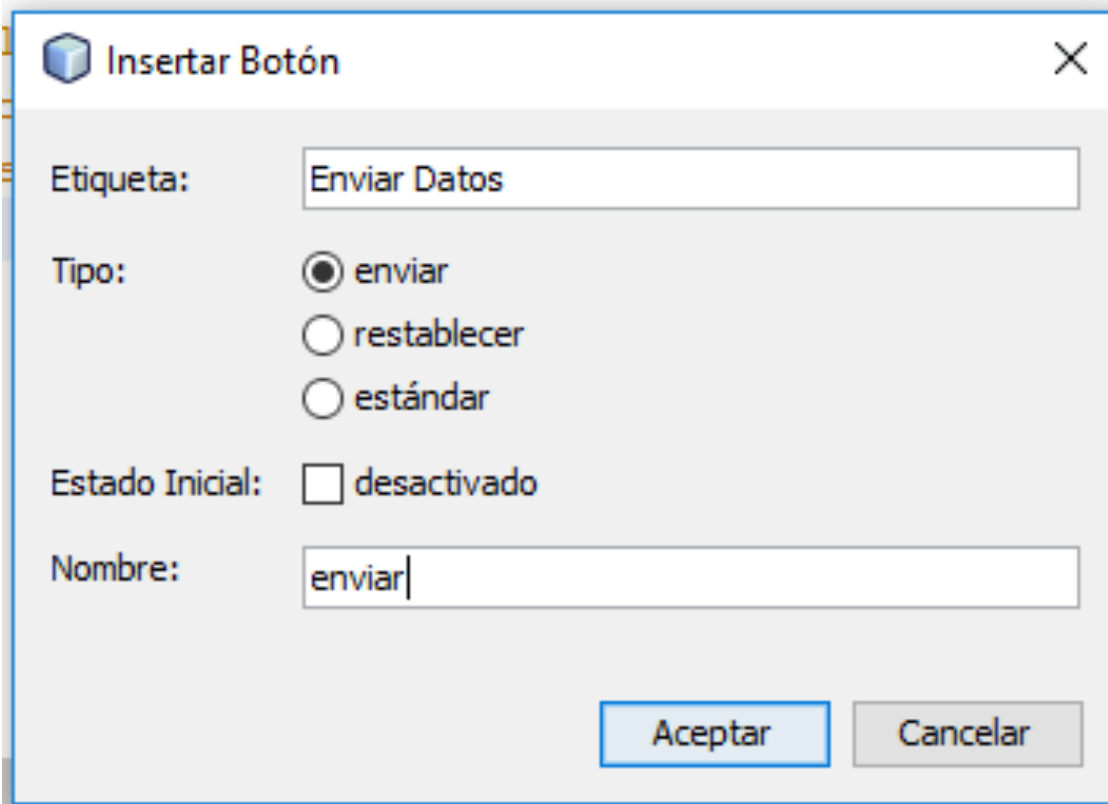
[php:](#)

```
<?php  
$enviar = $_request['enviar'];  
if (isset($enviar) == true)  
    echo 'se ha pulsado el botón de enviar';  
?>
```



Des de NETBEANS:

PALETA – BOTÓ. Tipus Enviar:

A screenshot of the 'Insertar Botón' (Insert Button) dialog box in NetBeans. The dialog has a title bar with a blue icon and a close button. It contains four labeled fields: 'Etiqueta:' with a text box containing 'Enviar Datos'; 'Tipo:' with three radio button options: 'enviar' (selected), 'restablecer', and 'estándar'; 'Estado Inicial:' with a checkbox labeled 'desactivado' which is unchecked; and 'Nombre:' with a text box containing 'enviar'. At the bottom right are two buttons: 'Aceptar' (highlighted with a blue border) and 'Cancelar' (disabled, in grey).

Insertar Botón

Etiqueta:

Tipo: ☒ enviar
☐ restablecer
☐ estándar

Estado Inicial: ☐ desactivado

Nombre:

SELECT SIMPLE

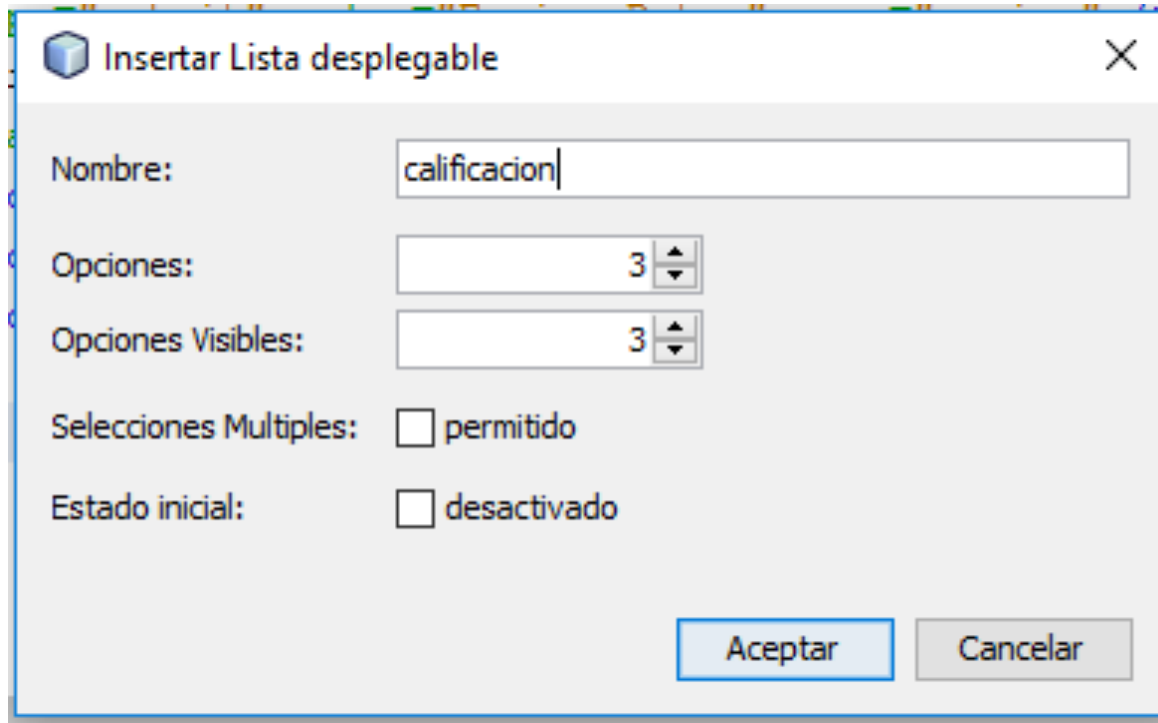
```
html:
calificación:
<select name="calificacion">
  <option value="nopresentado" selected>no presentado
  <option value="noapto">no apto
  <option value="apto">apto
</select>

php:
<?php
$calificacion = $_request['calificacion'];
echo $calificacion;
?>
```



Des de NETBEANS:

PALETA – LLISTA DESPLEGABLE:



Insertar Lista desplegable

Nombre:

Opciones:

Opciones Visibles:

Selecciones Múltiples: ☐ permitido

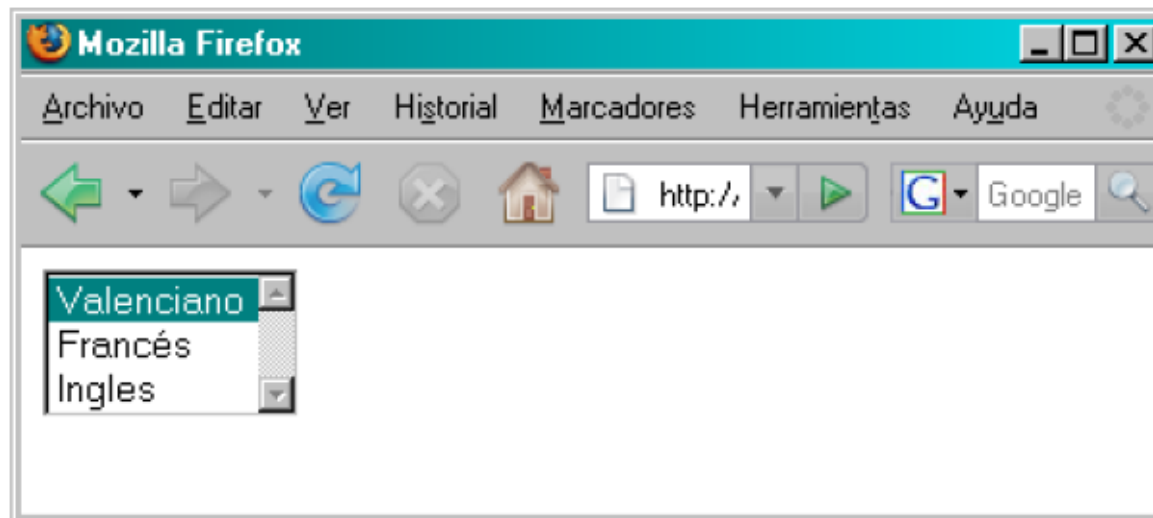
Estado inicial: ☐ desactivado

- Una vegada s'han creat les opcions, s'ha d'indicar nom de cada opció, mitjançant l'atribut *value* de option.
- Tambiés escriure el text que desitgem que es veja en cada opció. Estarà entre les etiquetes *<option>* i *</option>*

SELECT MÚLTIPLE

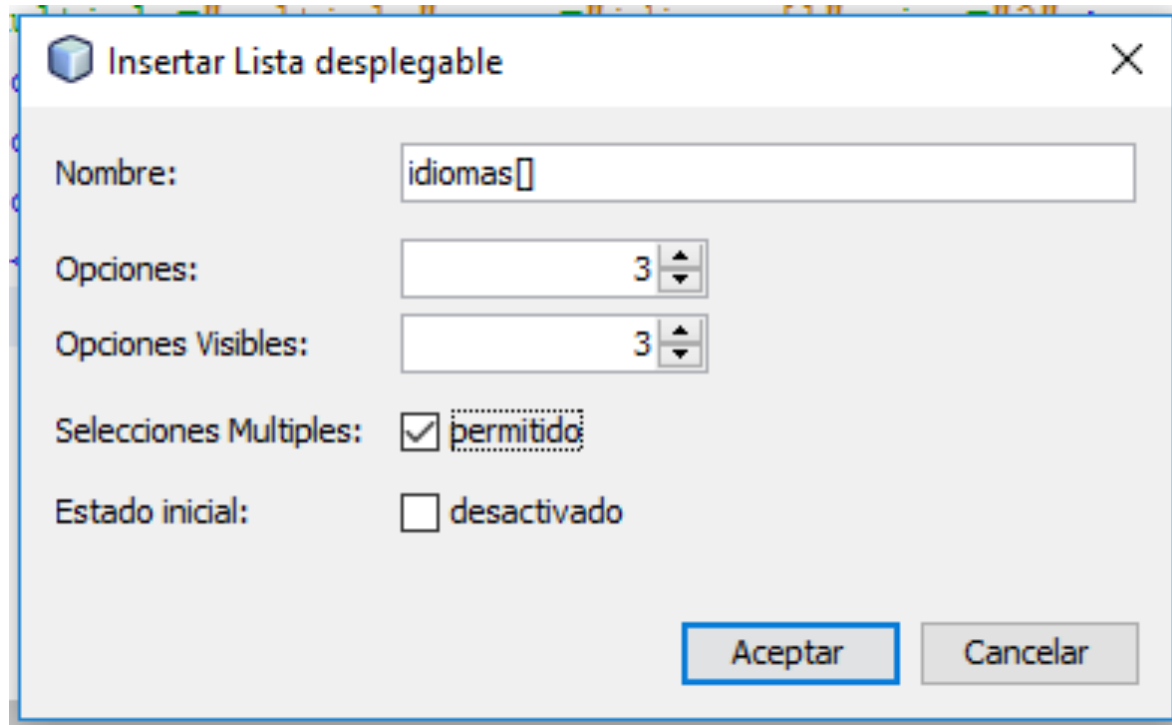
```
html:
<select multiple size="4" name="idiomas[]">
  <option value="valenciano" selected>valenciano
  <option value="frances">francés
  <option value="ingles">ingles
</select>

php:
<?php
$idiomas = $_request['idiomas'];
foreach ($idiomas as $idioma)
    echo '$idioma<br>';
?>
```



Des de NETBEANS:

PALETA – LLISTA DESPLEGABLE. Seleccions múltiples es deixa a Permés :

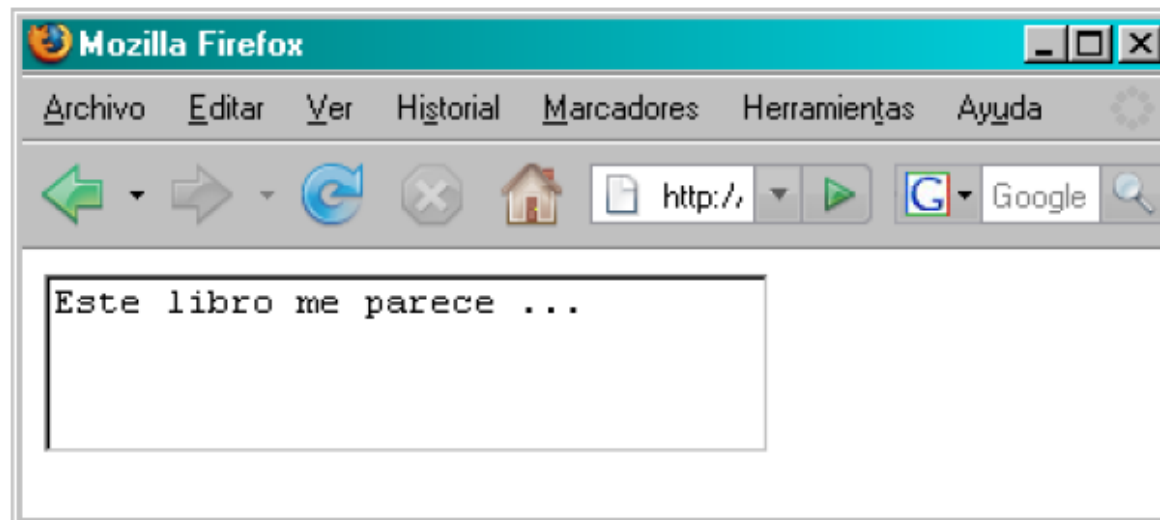


TEXTAREA

```

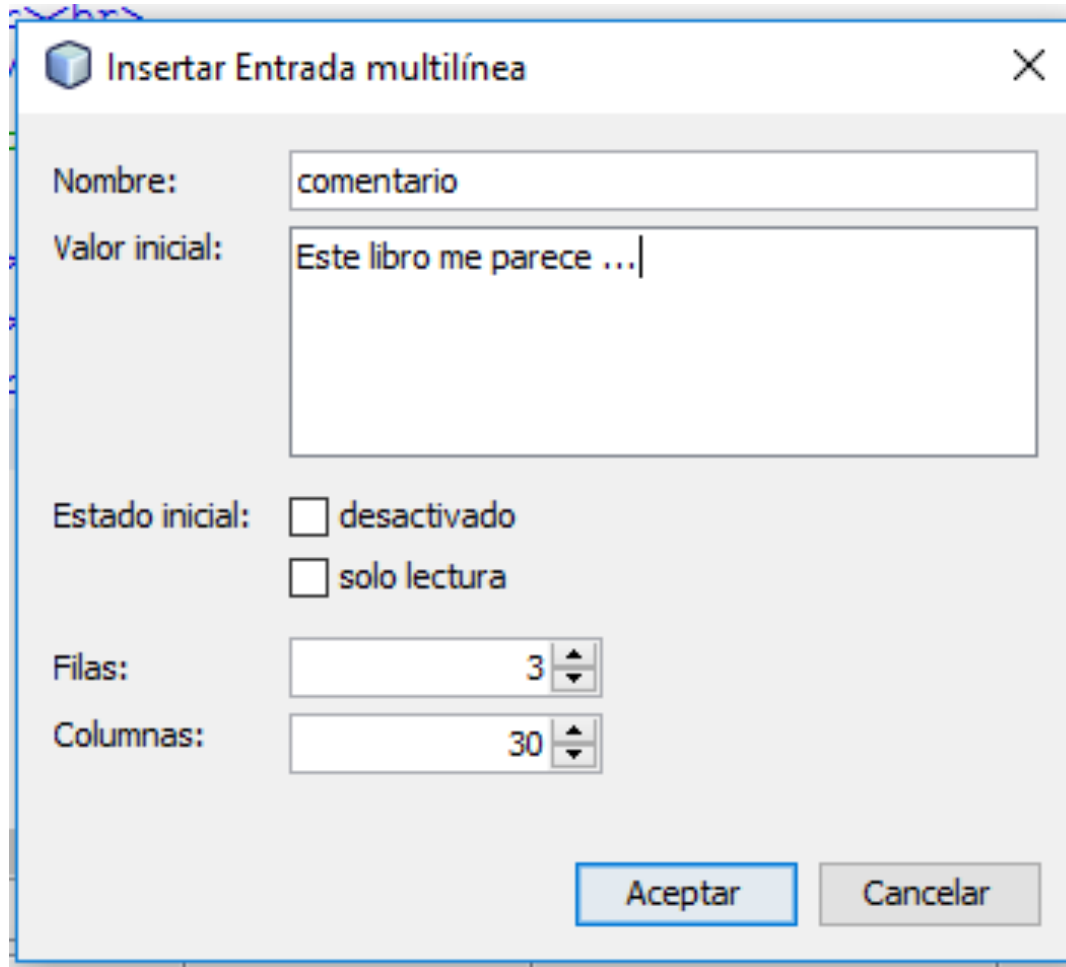
html:
comentario:
<textarea cols="30" rows="3" name="comentario">
este libro me parece ...
</textarea>

php:
<?php
$comentario = $_request['comentario'];
echo $comentario;
?>
    
```



Des de NETBEANS:

PALETA – ENTRADA MULTILÍNEA:



Insertar Entrada multilínea

Nombre:

Valor inicial:

Estado inicial: ☐ desactivado
☐ solo lectura

Filas:

Columnas:

INPUT IMAGE

Si volem enviar dades d'un formulari amb una imatge tindrem en compte que s'envien com a dada les coordenades en dos variables **Nom.x** i **Nom.i** que PHP transforma a **Nom_x** i **Nom_i** perquè siguin nom de variable vàlids.

```
html:
<form action="procesa_formulario.php" method="get">
<p><input type="text" name="texto" value="juan" size="20"></p>
<p><input type="image" width="75"
src="http://images.jupiterimages.com/common/detail/50/17/22851750.jpg"
name="envio" /></p>
</form>

php:
<?php
if (isset($_GET['envio_x']) == true)
    echo 'hola ' . $_GET['text'];
?>
```

En el *QUERY_STRING* es envia:

http://localhost:8084/proves/processa_formulari.php?cadena=Juan&envio.x=58&envio.i=46

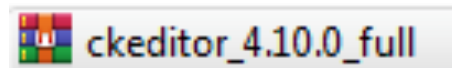


COM INCLOURE ELEMENTS DE FORMULARI DES DE CKEditor4:

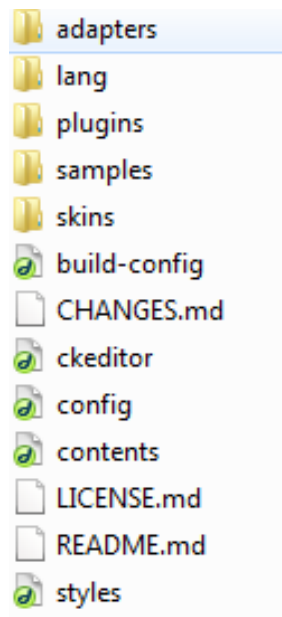
Descarregar eina des de:

<https://ckeditor.com/ckeditor-4/download/>

Descarreguem arxiu .zip:

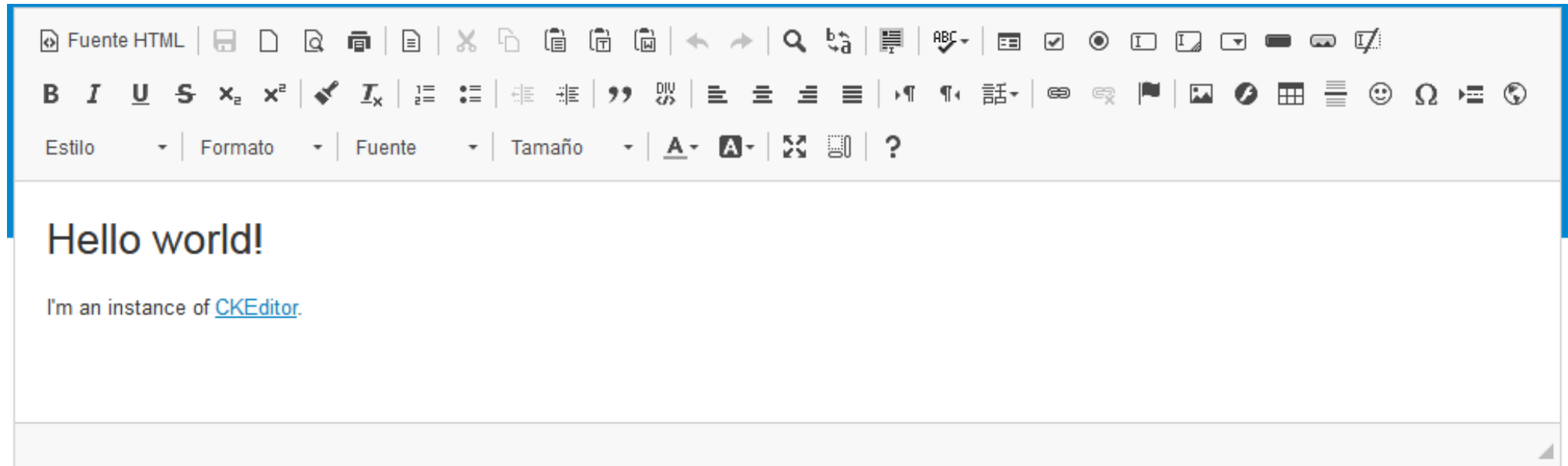


Ho descomprimim en qualsevol carpeta del nostre equip. Ens generarà la següent estructura de carpetes denrtó de la carpeta **ckeditor**:

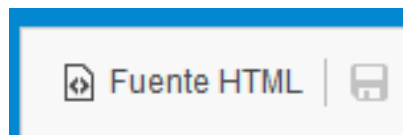


Per a executar entrem en **samples\index.html**.

Veurem aquesta pantalla:

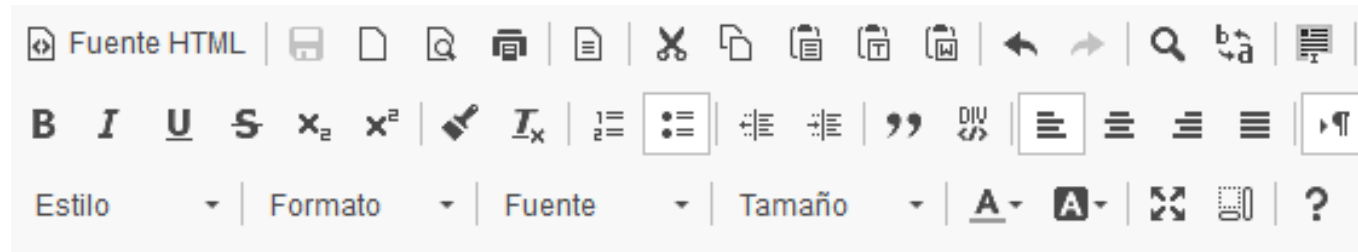


Ara anirem seleccionant les icones de cada tipus d'element html de la Barra d'Eines i, en prémer la pestanya



, veurem que codi html que s'ha generat.

Exemple: Escric diversos valors tipus vinyeta i li done format de llista no ordenada:



- blanco
- verde
- azul

Vegem l'html generat:



Afegir més exemples, de Formularis, Taules, etc.....

PAS DE DADES EN FORMULARIS

Les variables d'un script tenen una validesa exclusiva per al script i resulta impossible conservar el seu valor quan executem un altre arxiu diferent, encara que tots dos estiguen enllaçats.

1.2.1 Amb Variables Globals

- Cada nom de variable del formulari està disponible com a variable global (**\$nom**).
- Aquest és el sistema tradicional i el més simple d'implementar , però també **més insegur**, per la qual cosa es desaconsella.
- Només funcionarà si en l'arxiu *php.ini* la directiva *register_globals* està activada).

Exemple:

```
<input type="text" name="nom">
```

1.2.2 Amb Arrays Associatius

- **\$_POST**: Array associatiu que conté les variables passades pel mètode POST.
- **\$_GET**: Array associatiu que conté les variables passades pel mètode GET
- **\$_REQUEST**: Array associatiu que conté les variables passades pel mètode. REQUEST. Conté tota la informació de GET , POST i COOKIE.

Nota:

Com ja es va dir, `$_request[]` és una variable GLOBAL i és accessible des de qualsevol part del script. A partir de la PHP 5.4.0 només es pot accedir a dades del formularis HTML usant `$_POST[]` o `$_GET[]`.

La manera d'accedir als elements del formulari, i que són elements d'aquests arrays seria, `$_POST['nom']` on "*nom*" seria l'una variable del formulari i, per tant , equivalent a *\$nom*, tenint en tots dos casos el valor introduït en el formulari.

Nota:

L'element `REQUEST_METHOD` de la matriu global `$_SERVER` indica el mètode d'enviament .

Exemple:

```
if($_SERVER["REQUEST_METHOD"]=="POST")
    print_r($_POST);
else
    print_r($_GET);
```


1.2.3 Pas de variables per la url

Per a passar les variables d'una pàgina a una altra ho podem fer introduint aquesta variable dins del enllaç hipertext de la pàgina destí. La sintaxi seria la següent:

```
<a href="destí.php?variable1=valor1&variable2=valor2&..."> El meu enllaç</a>
```

- Aquestes variables no posseeixen el símbol \$ davant.
- Això és pel fet que en realitat aquesta manera de passar variables no és específic de PHP sinó que és utilitzat per altres llenguatges.
- Ara la nostra variable pertany també a l'àmbit de la pàgina *destí.php* i està llesta per a la seua utilització.

Nota:

No sempre es defineixen automàticament les variables rebudes per paràmetre en les pàgines web, depén d'una variable de configuració de PHP : *register_globals*, que ha de estar **activada** perquè així siga.

Exemple:

Tenim dues pàgines, *origen.html* i *destinació.php*:

origen.html

```
<html>
  <head>
    <title>origen.html</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body>
    <a href="destino.php?saludo=hola&texto=Esto es una variable texto">
      Paso variables saludo y texto a la página destino.php</a>
  </body>
</html>
```

destinació.php

```
<html>
<head>
<title>destino.php</title>
</head>
<body>
<?php
$saludo=$_GET["saludo"];
$texto=$_GET["texto"];

echo "contenido variable \$saludo: $saludo <br>\n";
echo "contenido variable \$texto: ".$texto." <br>\n"
?>
</body>
</html>
```

1.2.4 Pas de Matrius a PHP.

Ja hem vist en els casos de selecció múltiple (**SELECT MÚLTIPLE**) com enviar les dades en una matriu. Però ens pot interessar enviar diverses dades de diferents camps d'un formulari ficats en una **matriu** indexada.

Exemple:

Form_pasoMatriz.html:

```
<form action="form_pasoMatriz.php" method="post">
  <fieldset>
    <legend>datos propios</legend>
    <div>nombre:
    <input type="text" name="propio[nombre]" value="" size="20"></div>
    <div>correo:
    <input type="text" name="propio[correo]" value="" size="20"></div>
  </fieldset>
  <fieldset>
    <legend>datos conyuge</legend>
    <div>nombre:
    <input type="text" name="conyuge[nombre]" value="" size="20"></div>
    <div>correo:
    <input type="text" name="conyuge[correo]" value="" size="20"></div>
  </fieldset>
  <div>
    <input type="submit" name="enviar" value="enviar">
    <input type="reset" name="borrar" value="borrar formulario">
  </div>
</form>
```

Form_pasoMatriz.php

La funció *extract* permet associar una variable a una posició del array.

El consige enllaçant *nombreDeArray_nombrePosición*. En el meu cas amb *\$propi_nom*, és a dir usant en guió baix

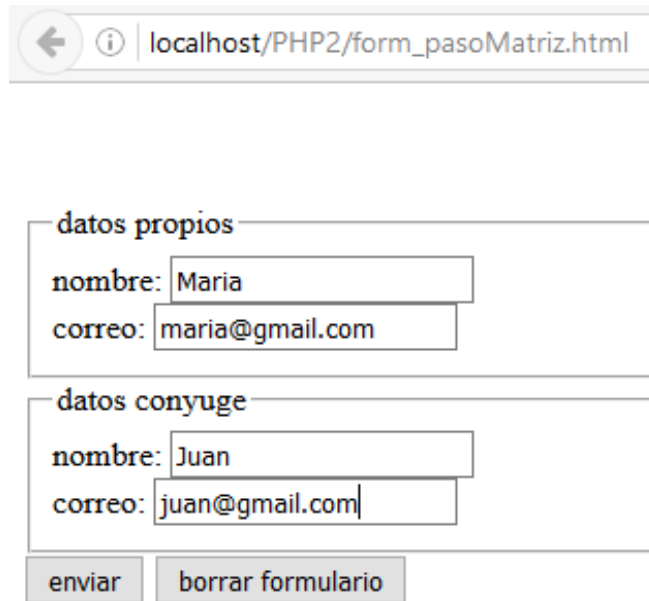
```
<?php
    extract($_POST['propio'] , EXTR_PREFIX_ALL, 'propio');
    extract($_POST['conyuge'] , EXTR_PREFIX_ALL, 'conyuge');
    echo '<table>';
        echo '<tr>';
            echo '<td></td><td>nombre</td><td>correo</td>';
        echo '</tr>';
        echo '<tr>';
            echo '<td>propio</td>';
            echo '<td>'.$propi_nombre.'</td>';
            echo '<td>'.$propi_correo.'</td>';
        echo '</tr>';
        echo '<tr>';
            echo '<td>conyuge</td>';
            echo '<td>'.$conyuge_nombre.'</td>';
            echo '<td>'.$conyuge_correo.'</td>';
        echo '</tr>';
    echo '</table>';
?>
```

Nota:

Veure informació de funció *extract()* en:

http://www.w3schools.com/php/func_array_extract.asp

Execució:



← ⓘ localhost/PHP2/form_pasoMatriz.html

datos propios

nombre:

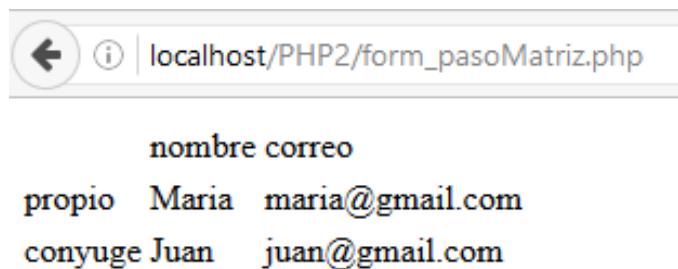
correo:

datos conyuge

nombre:

correo:

En prémer Enviar, s'obté:



← ⓘ localhost/PHP2/form_pasoMatriz.php

	nombre	correo
propio	Maria	maria@gmail.com
conyuge	Juan	juan@gmail.com

1.2.5 Pas de dades per QUERY_STRING.

A vegades interessarà passar dades a través d'un enllaç en lloc d'un formulari, per exemple per a passar variables de Javascript a PHP o per a passar un identificador de sessió. En aqueixos casos serem nosaltres els qui compondrem un QUERY_STRING amb les variables Javascript o l'aneu de sessió per a passar-los a través d'un missatge HTTP-GET.

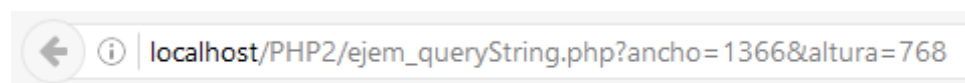
Veure exemples en : <http://www.devservice.es/blog/como-pasar-variables-javascript-a-php/>

Exemple:

ejem_QueryString.php:

```
<?php
// Como hemos pasado los datos en el QUERY_STRING usamos $_GET
if (isset($_GET['ancho'])===true && isset($_GET['altura'])===true){
    echo "El ancho de la pantalla es: ".$_GET['ancho']."<br/>";
    echo "La altura de la pantalla es: ".$_GET['altura']."<br/>";
}
else{
    echo "<script language='javascript'>\n";
    // Pasamos las variables al propio script.
    echo " location.href=\"${_SERVER['SCRIPT_NAME']}?\";
    // Concatenamos el QUERY_STRING que ya tuviera la pagina.
    echo "${_SERVER['QUERY_STRING']}";
    // Concatenamos vbles de Javascript alto y ancho de la resolución que está usando el usuario.
    echo "ancho=\" + screen.width + \"&\";
    echo "altura=\" + screen.height;\n";
    echo "</script>";
    echo '<a class="enlace" href="ejem_queryString.php?id='.$_SESSION['idSesion'].'">'.$texto_enlace.'</a>';
    exit();
}
```

Execució:



Ejemplo paso datos por Query String

El ancho de la pantalla es: 1366

La altura de la pantalla es: 768

1.2 PROCESSAMENT DE FORMULARIS EN PHP

1.3.1 Processament en un sol Script. FORMULARIS REENTRANTES

Fins ara s'ha vist l'esquema d'enviament de dades per formulari en què participen dues pàgines, una que conté el formulari i una altra que rep les dades d'aquest formulari.

El mateix ocorre quan enviem variables per una URL. Tenim una pàgina que contindrà l'enllaç i una altra pàgina que rebrà i tractarà aqueixes dades per a mostrar uns resultats.

En el present apartat veurem com es pot enviar i rebre dades d'un formulari amb una única pàgina.

A aquest efecte se'l denomina **autocrida de pàgines**, també se li sol cridar com a formularis **reentrantes** o termes similars. Se sol usar aquest esquema per a, per a formularis o enviament de dades per la URL

La manera de funcionament és:

- Si no es reben dades → Mostre el formulari o els enllaços que passen variables.
- Si rebo dades → Llavors he de processar el formulari o les variables de la URL

Avantatges:

- Disminueix el nombre de fitxers.
- Genera codi ordenat
- Permet validar les dades del formulari en el propi formulari.

Exemple1:

De formulari *reentrante*

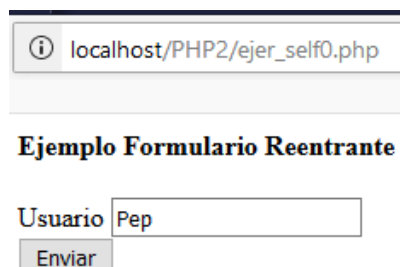
Ejer_self0.php:

```
<body>
<?php
if(isset($_POST['enviado']))
    if (!empty($_POST['name']))
    {
        $name = $_POST['name'];
        echo "Soy el usuario: <b> $name </b>";
        exit();
    }

?>
<h4>Ejemplo Formulario Reentrante</h4>
<form method="post" action="ejercicio_self0.php">
    <input type="text" name="name"><br>

    <input type="submit" name="enviado" value="Formulario"><br>
</form>
</body>
```

Execució

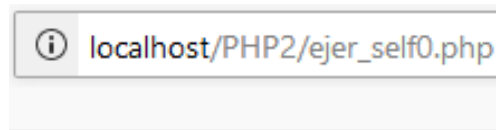


localhost/PHP2/ejer_self0.php

Ejemplo Formulario Reentrante

Usuario

En prémer Enviar veurem:

A screenshot of a web browser's address bar. It contains the text 'localhost/PHP2/ejer_self0.php' preceded by an information icon (a lowercase 'i' inside a circle). The address bar has a light gray background and a thin border.

Soy el usuario: **Pep**

En l'exemple , el primer pas és conéixer si s'estan rebent o no dades per un formulari. Per a això es comprova amb una sentència *if* si existeix o no una variable `$_POST`.

Nota:

if (!\$_POST) voldria dir alguna cosa com "*Si no existeixen dades vingudes d'un formulari*".

En cas que no existisquen, mostre el formulari. En cas que sí que existisquen, recull les dades i els imprimisc en la pàgina.

Una altra forma seria usant valor de la variable interna **`$PHP_SELF`**.

`$PHP_SELF` té tota la informació pertinent al servidor web on s'executa el script PHP, la ruta d'on s'executa, el nom de l'arxiu que s'executa... les variables GET i POST que se li envien a l'arxiu que s'executa i un sense fi més.

`$_SERVER['PHP_SELF']`

Veure manuals per a `$_SERVER` en:

<http://php.net/manual/en/reserved.variables.server.php>

Ejemplo2:

De formulari *reentrante*

Usar `$_SERVER['PHP_SELF']` en el **action** del formulari:

```
action="<?php echo $_SERVER['PHP_SELF']; ?>">
```

Ejer_self1.php:

```
if(isset($_POST['enviado']))
    if (!empty($_POST['name']))
    {
        $name = $_POST['name'];
        echo "Soy el usuario: <b> $name </b>";
        exit();
    }

?>
<h4>Ejemplo Formulario Reentrante</h4>
<form method="post" action="<?php echo $_SERVER['PHP_SELF']; ?>">
    <input type="text" name="name"><br>
    <input type="submit" name="enviado" value="Formulario"><br>
</form>
```

Ejemplo3:

De formulari *reentrante*

Sense usar exit();

Ejer_self2.php:

```

<body>
<?php
//if (!$_POST){ // es decir, si no enviamos nada desde el formulario
    if (empty($_POST['nombre'])
        || empty($_POST['empresa']) ||
        empty($_POST['telefono']))
    {
        // nota: el teléfono no será necesario, pues le doy valor por defecto
    }
?>

<form action="ejercicio2.php" method="POST">
    Nombre: <input type="text" name="nombre" size="30"><br>
    Empresa: <input type="text" name="empresa" size="30"><br>
    Telefono: <input type="text" name="telefono" size="14" value="+34"><br>
    <input type="submit" value="Enviar">
</form>

<?php
} else
{
    echo "<br>Su nombre: " . $_POST["nombre"];
    echo "<br>Su empresa: " . $_POST["empresa"];
    echo "<br>Su Teléfono: " . $_POST["telefono"];
}
?>
</body>

```

1.3.2 Validació de formularis des del servidor

REVISAR

Com ja esmentem, podrem validar les dades en el costat del client amb alguna mena de script o des del servidor en un procés d'anada i volta.

Per a validar les dades des del servidor es poden usar dos tipus d'esquema:

- Mostrar el formulari més d'una vegada:

```

SI Se ha enviado el formulario
  SI Hay Errores
    Mostrar formulario con errores
  SINO
    Procesar formulario
  FINSI
SINO Mostrar formulario
FINSI

```

- mostre el formulari només una vegada:

```

SI Se ha enviado el formulario
  Validar datos
  FINSI
SI Se ha enviado el formulario y no hay errores
  Procesar formulario
SINO
  Mostrar formulario con valores por defecto o ya enviados
  Mostrar algún aviso de error.
FINSI

```

Exemple del segon esquema:

Suposem que disposem de la següent biblioteca de funcions de validació ***valguda_dades.inc***:

Veure funció *ereg()* en :

<http://php.net/manual/es/function.ereg.php>

Veure expressions regulars en:

<http://stackoverflow.com/questions/1653425/a-za-z-a-za-z0-9-regular-expression>

valida_datoss.inc:

```
<?php
function valida_EsTexto($texto)
{
    $reg = "[A-Za-z]+[A-Za-z0-9]*$";
    return ereg($reg, $texto);
}
function valida_EsNumerico($texto)
{
    $reg = "[0-9]+$";
    return ereg($reg, $texto);
}
function valida_CamposCorrectos($datos, $tipos, &$errores)
{
    $numDatos = count($datos);
    assert($numDatos == count($tipos));
    $correctos = true;
    for ($i=0; $i < $numDatos; ++$i)
    {
        switch($tipos[$i])
        {
            case 'texto':
                $errores[$i] = !valida_EsTexto($datos[$i]);
                break;
            case 'entero':
                $errores[$i] = !valida_EsNumerico($datos[$i]);
                break;
            default: assert(false);
        }
        if ($errores[$i] === true)
            $correctos = false;
    }
    return $correctos;
}
?>
```

valida_datosServidor.php

```
html>
<head>
<title>ejemplo validacion</title>
<style>
<!--
.stynormal { font-family: arial; font-size: 10pt; }
.styerror { color: #ff0000; font-family: arial; font-size: 10pt; font-
style: italic }
-->
</style>
</head>
<body>
<?php
include_once('valida.inc');

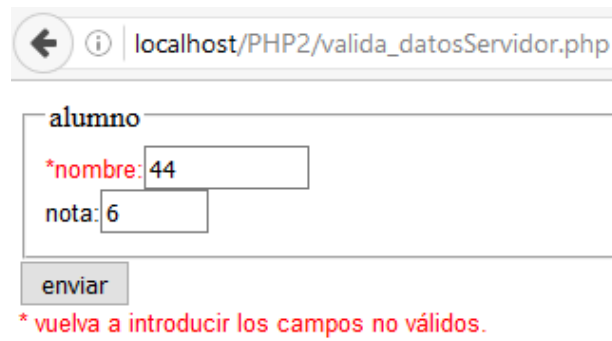
$campos = array('nombre','nota');
$tipos = array('texto','entero');
$errores = array(false, false);
$enviadoformulario = isset($_POST['datosenviados']);

if ($enviadoformulario === true)
{
    $datos = $_POST['datos'];
    if (valida_camposcorrectos($datos, $tipos, $errores))
        $hayerroresenformulario = false;
    else
        $hayerroresenformulario = true;
}
else
{
    $datos = array("", "");
    $hayerroresenformulario = false;
}
```

```

if ($enviadoformulario === true && $hayerroresenformulario === false)
{
    $numdatos = count($datos);
    for ($i=0; $i < $numdatos; ++$i)
        echo '<div>'.$campos[$i].': '.$datos[$i].'/>';
}
else
{
    echo '<form action="'.$_SERVER['PHP_SELF'].'" method="post">';
    echo '<fieldset>';
    echo '<legend>alumno</legend>';
    echo '<div class="';
    echo $errores[0]?'styerror">*: 'stynormal">';
    echo $campos[0].': '";
    echo '<input type="text" name="datos[]" value="'. $datos[0].'"';
        size="10"></div>';
    echo '<div class="';
    echo $errores[1]?'styerror">*: 'stynormal">';
    echo $campos[1].': '";
    echo '<input type="text" name="datos[]" value="'. $datos[1].'"';
        size="5"></div>';
    echo '</fieldset>';
    echo '<input type="submit" name="datosenviados" value="enviar">';
    echo '</form>';
    if ($hayerroresenformulario === true)
        echo '<div class="styerror">* vuelva a introducir los campos no válidos.</div>';
}
?>
</body>
</html>
    
```


Execució amb dades incorrectes:



localhost/PHP2/valida_datosServidor.php

alumno

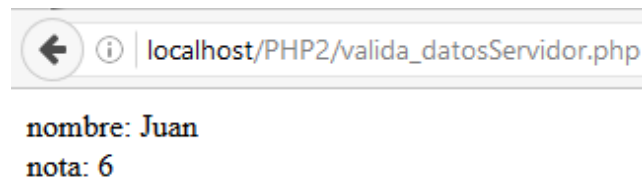
*nombre: 44

nota: 6

enviar

* vuelva a introducir los campos no válidos.

Execució amb dades correctes:



localhost/PHP2/valida_datosServidor.php

nombre: Juan

nota: 6

6. ENLLAÇOS D'INTERÉS

Tutorial HTML:

<http://www.htmlquick.com/es/tutorials.html>

Formularis:

<http://www.htmlquick.com/es/tutorials/forms/4.html#practice>